

CLAIMS

I claim:

- 1 1). A method, comprising:
 - 2 analyzing each routine, of a software program having a plurality of separately
 - 3 compilable routines, to create a plurality of local side-effect problems for
 - 4 each routine; and
 - 5 merging the local side-effect problems to create a global side-effect problem.
- 1 2). The method of claim 1, further comprising:
 - 2 computing a global solution to the global problem; and
 - 3 splitting the global solution into local solutions.
- 1 3). The method of claim 2, further comprising:
 - 2 determining for each routine, whether a pointer parameter within the routine
 - 3 is used to write to or read from a storage device.
- 1 4). The method of claim 3, further comprising:
 - 2 determining for each routine whether the pointer parameter is used to derive
 - 3 a return value of the routine.
- 1 5). The method of claim 4, further comprising:
 - 2 computing a lattice value associated with each of the pointer parameters,
 - 3 wherein the lattice value comprises one of a PURE effect; LOST effect;
 - 4 RETURN effect; OUT effect; IN effect; RETURN, OUT, and IN effect;
 - 5 RETURN and OUT effect; RETURN and IN effect; and OUT and IN effect.
- 1 6). The method of claim 5, further comprising:
 - 2 providing the lattice values to an interprocedural analysis solver to optimize
 - 3 compilation of the software program.

4 determining for each routine, whether a pointer parameter within the routine
5 is used to write to or read from a storage device.

1 11). The computer-readable medium of claim 10 having stored thereon
2 additional instructions, said additional instructions when executed by a
3 computer, cause said computer to further perform:
4 determining for each routine whether the pointer parameter is used to derive
5 a return value of the routine.

1 12). The computer-readable medium of claim 11 having stored thereon
2 additional instructions, said additional instructions when executed by a computer,
3 cause said computer to further perform,
4 computing a lattice value associated with each of the pointer parameters,
5 wherein the lattice value comprises one of a PURE effect; LOST effect;
6 RETURN effect; OUT effect; IN effect; RETURN, OUT, and IN effect;
7 RETURN and OUT effect; RETURN and IN effect; and OUT and IN effect.

1 13). The computer-readable medium of claim 12 having stored thereon
2 additional instructions, said additional instructions when executed by a computer,
3 cause said computer to further perform:
4 providing the lattice values to an interprocedural analysis solver to optimize
5 compilation of the software program.

1 14). The computer-readable medium of claim 13 having stored thereon
2 additional instructions, said additional instructions when executed by a
3 computer, cause said computer to further perform:
4 representing the local side-effect problems as directed graphs having edges
5 and vertices, wherein
6 each edge has an associated monotone transfer function;

7 each vertex has a vertex value, wherein the vertex value is one of formal
 8 parameter, implicit parameter, local pointer variable, or gate
 9 parameter; and
 10 a subset of vertices is marked with lattice values.

1 15). A system, comprising:
 2 a processor;
 3 memory connected to the processor storing instructions for interprocedural
 4 side-effect analysis executed by the processor;
 5 storage connected to the processor that stores a software program having a
 6 plurality of separately compilable routines,
 7 wherein the processor analyzes each routine, of the software program, to
 8 create a plurality of local side-effect problems for each routine; and
 9 merges the local side-effect problems to create a global side-effect
 10 problem.

1 16). The system of claim 15, wherein the processor computes a global solution
 2 to the global problem; and splits the global solution into local solutions.

1 17). The system of claim 16, wherein the processor determines for each
 2 routine, whether a pointer parameter within the routine is used to write to or
 3 read from the storage device.

1 18). The system of claim 17, wherein the processor determines for each
 2 routine whether the pointer parameter is used to derive a return value of the
 3 routine.

1 19). The system of claim 18, wherein the processor:

2 computes a lattice value associated with each of the pointer parameters,
3 wherein the lattice value comprises one of a PURE effect; LOST effect;
4 RETURN effect; OUT effect; IN effect; RETURN, OUT, and IN effect;
5 RETURN and OUT effect; RETURN and IN effect; and OUT and IN effect.

1 20). The system of claim 19, wherein the processor:
2 provides the lattice values to an interprocedural analysis solver to optimize
3 compilation of the software program.

1 21). The system of claim 20, wherein the processor:
2 represents the local side-effect problems as directed graphs having edges
3 and vertices, wherein
4 each edge has an associated monotone transfer function;
5 each vertex has a vertex value, wherein the vertex value is one of formal
6 parameter, implicit parameter, local pointer variable, or gate
7 parameter; and
8 a subset of vertices is marked with lattice values.